

ё

G-MD2-1 «Платформа Атом.РИТА»
Модуль Атом.РИТА: Классификатор
Руководство пользователя

На 11 листах

Дата: 06.04.2022

Версия: 1.0.0

Лист изменений

| Дата | Версия | Краткое описание изменений | ФИО |
| --- | --- | --- | --- |
| 06.04.2022 | 1.0.0 | Документ создан | Поляков А.В. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Содержание

[Глоссарий 4](#_Toc100136224)

[1. Введение 5](#_Toc100136225)

[1.1. Назначение системы 5](#_Toc100136226)

[1.2. Документация, с которой необходимо ознакомиться пользователю 5](#_Toc100136227)

[2. Требования к пользователю 6](#_Toc100136228)

[3. Состав платформы 7](#_Toc100136229)

[4. Работа с модулем 8](#_Toc100136230)

[4.1. API модуля Атом.РИТА: Классификатор 8](#_Toc100136231)

[4.2. Сценарий взаимодействия роботов с модулем Атом.РИТА: Классификатор 9](#_Toc100136232)

[5. Аварийные ситуации 11](#_Toc100136233)

Глоссарий

| **Термины/сокращения** | **Определение** |
| --- | --- |
| Платформа Атом.РИТА, Платформа | Роботизированный интеллектуальный технологичный ассистент |
| Программный робот (Робот) | Программное обеспечение для эмуляции действий человека, взаимодействующего с информационными системами, для выполнения типового бизнес-процесса |

# Введение

## Назначение системы

Платформа Атом.РИТА предназначена для разработки и управления программными роботами, позволяющими автоматизировать бизнес-процессы путем воспроизведения действий пользователей в установленных на рабочих станциях программах. Данный документ описывает модуль Атом.РИТА: Классификатор из состава Платформы Атом.РИТА. Состав Платформы подробно рассмотрен в разделе 3 настоящего документа.

## Документация, с которой необходимо ознакомиться пользователю

В зависимости от конфигурации Платформы, может потребоваться ознакомление с Руководством пользователя модуля Атом.РИТА: Разработчик.

# Требования к пользователю

Пользователь должен знать и соблюдать установленную технологию обработки информации, выполнять руководства, инструкции и положения по обеспечению защиты информации. Ответственным за ознакомление Пользователей c руководствами, инструкциями и положениями по обеспечению защиты информации является Администратор информационной безопасности (далее – Администратор ИБ)

Обо всех выявленных нарушениях, связанных с информационной безопасностью, Пользователь должен незамедлительно сообщать Администратору ИБ.

# Состав платформы

Платформа Атом.РИТА предназначена для разработки и управления программными роботами, позволяющими автоматизировать бизнес-процессы путем воспроизведения действий пользователей в программах, установленных на рабочих станциях.

 Платформа состоит из пяти модулей (см. рисунок 1):

* Атом.РИТА: Разработчик – модуль, предназначенный для создания, хранения, изменения и отладки программных роботов, реализованный как веб-приложение.
* Атом.РИТА: Администратор – модуль, предназначенный для автоматического запуска и управления роботами, созданными в модуле Атом.РИТА: Разработчик, а также мониторинга их состояния и аудита работы, реализованный как веб-приложение.
* Атом.РИТА: Сотрудник – модуль, предназначенный для выполнения программного робота, устанавливаемый на рабочую станцию.
* Атом.РИТА: Классификатор - модуль, предназначенный для автоматического определения классов (признаков, характеристик) текстовых объектов. Модуль возможно обучить для работы с различными текстами и определения выбранных пользователем классов (от 1 до 5).
* Атом.РИТА: Голос – модуль, предназначенный для распознавания речи и ее перевода в текст, а также синтеза речи на основании заданного текста.



Рисунок 1. Функциональная структура платформы Атом.РИТА

# Работа с модулем

## API модуля Атом.РИТА: Классификатор

**Endpoint:**core-s-taiweb01.gk.rosatom.local:13089/rmqgateway/

**Метод:** Get

**Входные параметры**:

|  |  |
| --- | --- |
| short | textкраткое описание запроса |
| full | textполное описание запроса (может быть идентичным краткому) |
| org | не заполняем (всегда пустой) |
| loc | не заполняем (всегда пустой) |

**Пример:**{"org":"","loc":"","short":"Не работает монитор","full":"Не работает монитор"}

**Логика:**

При поступлении нового сообщения модуль Атом.РИТА: Классификатор:

* определяет значения классов объекта;
* определяет похожие объекты, находящиеся в обучающей выборке;
* отправляет ответное сообщение с результатами классификации;

**Ответ:**

Response Schema: application/json

**Пример:**

{'strJSON':

'{"prediction": {"sd": "SD10829435",

"variants": [{"prob": "0.46", "class": "1", "escalation": "1"}, {"prob": "0.22", "class": "0", "escalation": "0"}, {"prob": "0.17", "class": "3", "escalation": "0"}, {"prob": "0.12", "class": "2", "escalation": "0"}, {"prob": "0.03", "class": "4", "escalation": "0"}],

"simular": [{"class": "2", "prob": 0.7027027027027027, "sd": "9"}, {"class": "2", "prob": 0.5172413793103449, "sd": "12"}, {"class": "1", "prob": 0.5, "sd": "1"}, {"class": "3", "prob": 0.4090909090909091, "sd": "16"}, {"class": "1", "prob": 0.40816326530612246, "sd": "6"}]}}'}}

**Входные параметры**:

|  |  |
| --- | --- |
| sd | string (sd)Идентификатор запроса |
| variants | Варианты ответа в порядке убывания вероятности |
| prob | Вероятность данного класса |
| class | Порядковый номер класса,определенного моделью |
| escalation | При превышении порога вероятности, первый вариант считается уверенным ответом и данный параметр равен у него единице. |
| similar | Перечисление похожих обращений из исторической выборки |
| class | Класс похожего обращений |
| prob | Процент сходства данного обращения |
| sd | id данного обращения |

Успех - код 200.

Ошибка валидации - код 422.

## Сценарий взаимодействия роботов с модулем Атом.РИТА: Классификатор

Пример сценария настройки робота в модуле Атом.РИТА: Разработчик для взаимодействия с модулем Атом.РИТА: Классификатор:

1. Создать проект робота.
2. В проект робота добавить активность/набор активностей для получения текста, который необходимо классифицировать: например, "Открыть Excel" (Open Excel) + "Прочитать ячейку" (Read Cell) или "Получить письма" (Get outlook mail messages)+"Присвоить значение" (Assign) для извлечения текста из темы/тела письма
3. Добавить активность "Выполнить python код" (Invoke Python Code) для формирования тела запроса в заданном виде:

import json

json\_body = json.dumps({
    'org': '',
    'loc': '',
    'short': text,
    'full': text
})

где: short, full - текст для классификации.

1. Добавить активность "HTTP Запрос" (Http Request) для отправки текста в модуль Атом.РИТА: Классификатор. Данная активность настраивается следующим образом:
* repeats\_num: 3;
* delay\_btw\_requests: 2000ms;
* endpoint: http://1tgrp-s-grp01.gk.rosatom.local:9024/predict;
* method: GET;
* auth\_type: NONE;
* body\_format: application/json;
* body: переменная с телом запроса (сформированная ранее, на шаге 3);
* accept\_format: JSON;
* result\_body: переменная типа base.string для сохранения ответа от сервиса Классификатора (ответ поступает синхронно);
* status\_code: переменная типа base.integer для сохранения кода состояния (ответа от сервиса Классификатора).
1. Проверяется ответ от модуля Атом.РИТА: Классификатор с помощью активности "Условие TRUE/FALSE": если ответ =200, то выполняются следующие шаги. В случае другого ответа - направляться ошибка в лог/ответственному лицу.
2. Добавить активность "Десериализовать JSON" (Deserialize JSON) для преобразования ответа в виде строки (от модуля Классификатор) в объект JSON.
3. Для полученного объекта JSON с помощью активности "Присвоить значение" получить необходимые данные из результата, например: значение ключа "class" (порядковый номер класса) для объекта, в ключе "escalation" которого указана "1" (устанавливается для класса с наибольшей вероятностью, если данная вероятность превышает порог).
4. Далее, в зависимости от проекта, добавить активность/набор активностей для или отправки определенного номера класса в необходимый файл/ответственному лицу/смежной системе.

# Аварийные ситуации

При отказе или сбое в работе модуля необходимо обратиться к Администратору или в службу технической поддержки.